ACEEE Int. J. on Information Technology , Vol. 3, No. 3, Sept 2013

Temporal Knowledge, Temporal Ontologies, and Temporal Reasoning for Managerial Tasks – an Attempt for a Survey

Maria Mach-Król
University of Economics, Katowice, Poland
maria.mach-krol@ue.katowice.pl

Abstract - **The paper is devoted to a variety of solutions aiming at taking time into account in economic and managerial analyses. It presents an example of a decision problem – namely the analysis of economic environment and of barriers to entry to a market space. Next, it shows how time can be treated in intelligent decision systems, in context of temporal knowledge representation and temporal reasoning.**

***Index Terms*: temporal reasoning, time, temporal intelligent systems, ontology, economic knowledge, economic environment**

I. Introduction

A modern economic domain may be called a turbulent one. Changes appear very quickly, and it is difficult to take them all into account. At the same time, if managerial decisions are to be correct, it is absolutely necessary to use knowledge about these changes. This in turn brings the need of using the temporal aspects of the environment, as changes occur in time. Therefore, economic knowledge is of temporal character, and reasoning about it also is temporal. Time becomes a crucial dimension of economic analysis.

The economic decisions are nowadays so complex, that they have to be supported by intelligent tools.

The goal of the paper is to present a broad survey of questions linked with dynamic analysis of enterprise's economic environment. Therefore we first discuss how time is dealt with in economics and in AI (Section 2). In Section 3 we present temporal systems and temporal knowledge together with their characteristics. Section 4 is devoted to the temporal conceptual graphs and their use in depicting temporal economic knowledge. We present an economic example of barriers to entry to a market space, in Section 5. We show how they can be described and analyzed using temporal conceptual graphs. As one of the main elements of temporal reasoning process is ontology, we discuss ontology of time for economic domain in Section 6. We show our proposal of a temporal ontology, mixing point time and calendar time. In Section 7 ontology representation in RDF and OWL is presented. An example of temporal reasoning for economic problems is presented in Section 8, together with the TAL language and the VITAL system. Finally in Section 9 we present so-called temporal reasoners, and discuss a solution proposed by the authors of the PROTON reasoner.

II. Time in Management and in Artificial Intelligence

If today enterprises operating on a free market, want to survive, become successful and to develop themselves, they have to react to changes in the economic environment quickly and properly. This concerns past, present and anticipated changes. Perceiving changes in this way is connected with a modern trend in strategic management, according to which enterprises treat analysis of anticipated changes and the reaction to present changes as equally important, very often the first type of analysis becomes more important.

At the same time, as Ansoff claims, in the environment of an enterprise there can be seen bigger and bigger turbulences, resulting from – among other factors – four processes: newness of change increase, pace of change increase, environment complexity increase, environment intensity increase [17].

It becomes obvious that in the strategic analysis we have to consider the time factor, being a key one. There exist many theories putting emphasis on time in strategic analysis. Here we will recall two of them, namely a concept of time-based competition and a conception called "economy of speed".

In the first theory "overtaking" the competitors in time is perceived as the main factor in enterprise's success. To do so, an enterprise needs research concerning the dynamics of phenomena and of processes in its economic environment as well as inside the firm. The second theory stresses the need for real-time management, which allows to start strategic operations before competitors start them, gaining in this way a so-called first-mover advantage.

The pace of changes may be so great that may cause problems to be solved in ad hoc manner. If the problems are to be solved by an intelligent system, also the system's knowledge has to be placed in temporal context, as it has to be always up to date. And one of the main reasons for knowledge changes is the passage of time.

All that has been said above leads to a simple conclusion: a good representation of economic knowledge should be a time-based one.

It is commonly known and accepted, that time representation and temporal reasoning are necessary in many AI systems, as time is the basis for reasoning about action and change. Many AI systems concern enterprise's environment which is constantly changing (see e.g. [20]).

Therefore, if decisions made on the basis of such systems' advices are to be correct, the system has to deal with

DOI: 01.IJIT.3.3.5

temporal dimension of information, changes of information in time as well as it should "possess" knowledge about the nature of those changes. The tasks for such temporal AI system encompass among others:

- maintaining temporal coherence,
- answering temporal queries,
- explanations,
- prediction, etc.

A temporal intelligent system is a system, that performs temporal reasoning explicitly. That is, the system not only contains e.g. fact base, a rule base, and an inference engine, but deals with the question of time directly. Such a system allows for inference about changes of phenomena in time, for historical analysis of phenomena, for prediction and– generally speaking – for a dynamic analysis of reality depicted.

The most common way of representing temporal knowledge is to use notations based on logic. Many authors have been dealing with this question and there are several temporal logic systems, which can be roughly divided into: logics of time intervals, logics of time points and logics of time points and intervals (see e.g. [10]).

There can be seen a real need for constructing temporal intelligent systems in areas of economy and management. As the examples of such areas one could point out:

- adaptive formulation of marketing strategy, depending on changes in competitive environment conditions,
- adapting investing strategy to economic and/or legal conditions,
- adapting development strategy of an enterprise to entry barriers on a relevant market, etc.

There have been proposed in the literature several solutions to adaptive temporal intelligent systems, which encompass, among others, the use of temporal logic, dynamic models (the ones based on states and predicative ones). Reiter [27] proposed to use situation calculus.

III. Temporal Systems and Temporal Knowledge

By a temporal intelligent system we understand an artificial intelligence system that explicitly and directly performs temporal reasoning (see Section 1 for definition). For an intelligent system to be considered temporal, explicit time references should be found at least in the representation and reasoning layers. A sample structure of a temporal intelligent system is presented in Fig. 1.

Temporal intelligent system may be considered a decision support system in a sense that its purpose is to support e.g. decisions that need an advanced temporal analysis of the economic environment. Taking into account the temporal features of the environment, we may see the tasks of the system from two different perspectives.

In an economic-managerial perspective, the intelligent analysis of changes encompasses:

1. a proper representation of phenomena in the economic environment and of their changes in time,
2. representation of both qualitative and quantitative phenomena,
3. analysis of the current state of the environment,
4. historical analysis of changes in the environment (tracing changes' evolution),
5. expression of causal relationships,
6. analysis of future changes in the environment.

A tool for performing the above tasks should be able to (instrumental perspective):

1. represent time – discrete and/or dense one, depending on the needs and on the nature of phenomena being analyzed,
2. represent causal relationships between actions and/or phenomena in the environment, represent conditions to perform an action,
3. represent processes, among which are conditional and concurrent ones,
4. analyze future changes.

What are the advantages of using temporal formalisms?

First of all, they allow for representing changes, and reasoning about causes, effects and directions of these changes. Next, with causal relationships, temporal formalisms may be used for "if-then" analysis. These formalisms provide representation for qualitative and quantitative temporal information, as well as temporal relations. It is possible to perform qualitative reasoning and to simulate human commonsense reasoning. With temporal formalisms it is possible to differ reasoning granulations, and to model processes in an explicit way.

Temporal knowledge is a special kind of knowledge. It is distinguished by the following properties:

- Time-dependent
- changing,
- encompassing changes and actions,
- encompassingcausalrelationships.

A temporal knowledge may be used in a temporal intelligent system. A system possessing such a knowledge is capable of gaining new knowledge, of providing up-to-date knowledge, of dealing with new information. As a result, it is possible in such system to represent changing domains, to fit into changes in system's environment and to reason about changing domains.

The temporal knowledge is not homogeneous. One can distinguish several types of it, depending on the scope of temporal assignments:

- Staticknowledge: *An enterprise has to conform to legal rules"*
- Sequences: *Passing a law ->signing the law by the President ->publishing the law*
- Temporallystampedknowledge: *Application for license ->decision ->valid period of license*
- Fullytemporalknowledge: *Varyingprices of shares.*

For a temporal intelligent system to use temporal knowledge, it has to be represented properly. In Sections 4 and 5 the question of knowledge representation is addressed, using the example of conceptual graphs.

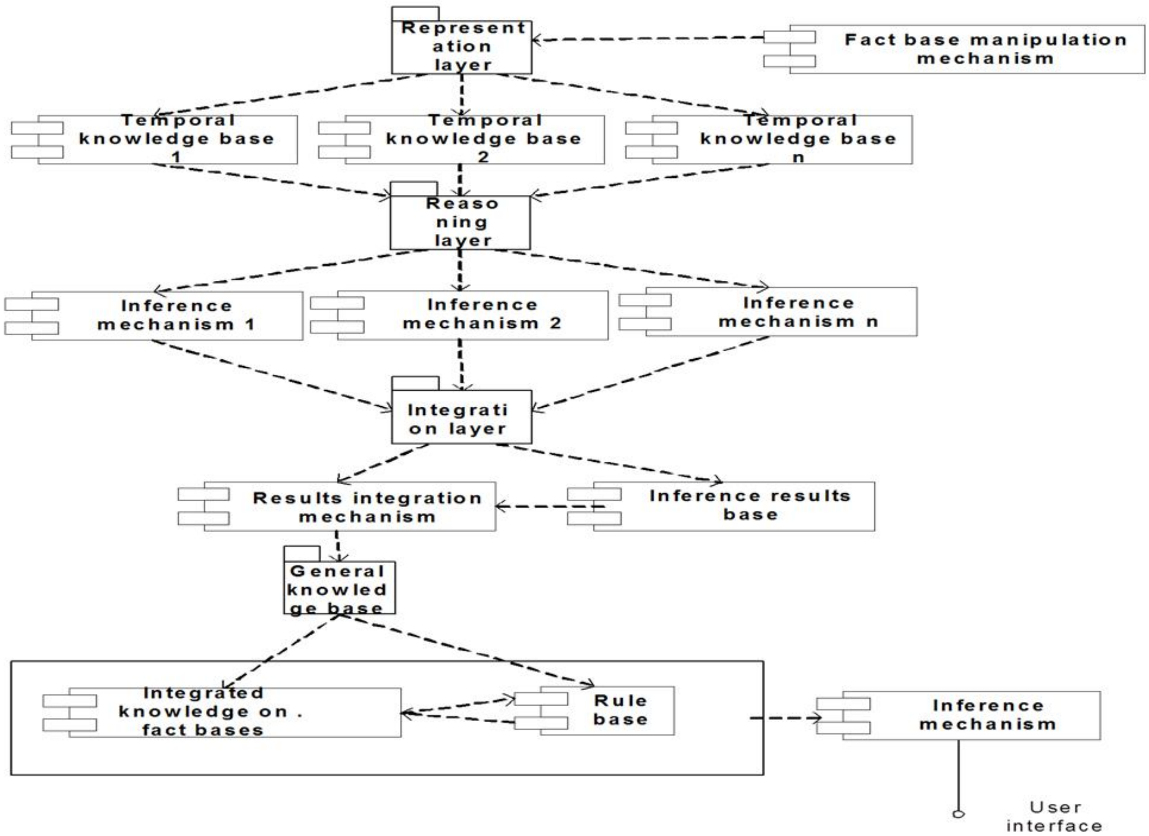

Figure. 1. A structure of a sample temporal intelligent system. Source: own elaboration

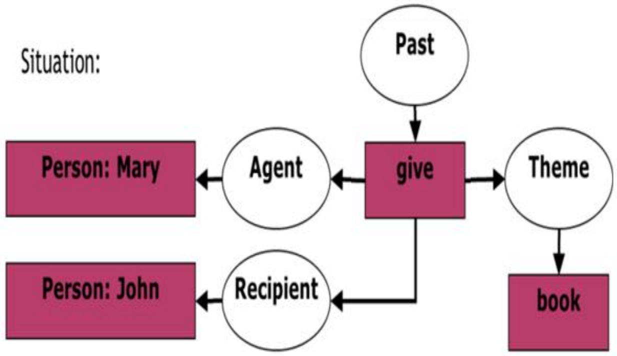

Figure 2. Sample conceptual graph.Source: own elaboration.

IV. Temporal Conceptual Graphs

Representation of knowledge means an attempt to make knowledge understandable to humans and machines. According to Sowa [29] knowledge representation is composed of: underlying logic, ontology and computational models.

By the knowledge representation we can also mean using logic and ontology to build computational models of a domain (ibid.)

One of the oldest knowledge representations is the one of semantic nets. It is a graphical notation.

Semantic nets have many variants, one of them is the idea of conceptual graphs by Sowa. Fig. 2 is an example of a simple conceptual graph.

A conceptual graph is a bipartite, directed graph with two kinds of nodes: notions and relations. It may be encoded graphically, with rectangles representing notions and ovals representing relations; or as text, where notions are in square brackets and relations are in parenthesis. Conceptual graphs have many advantages, among others:

- They are legible, easy to understand, and at the same time it is a strictly formal notation,
- They are useful for representing AI problems, e.g. formalizing natural language sentences,
- They are easily expressed in KIF[1] and in first order logic, which allows to use CG rules in many types of reasoning systems.

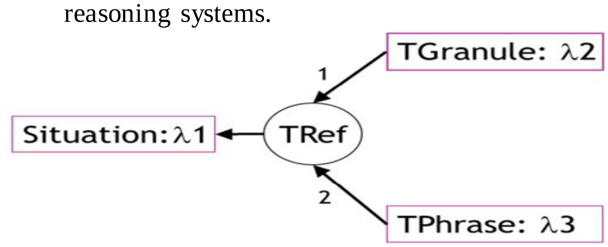

Figure 3. Simple temporal annotations in a CG.Source: after[16].

A drawback of CG notation is that it is not adapted to formalization of temporal phenomena. Therefore in temporal systems it is needed to augment the CG formalism with temporal qualifications, taking into consideration the characteristics of the domain and of the reasoning system. Simple schema of temporal annotations in a conceptual graph are presented in fig. 3.

[1] KIF – Knowledge Interchange Format.

- TGranule = [TZone: λ1] → (TZne) → [TLabel: λ2] (TGty) ← [TGranularity: λ3]
- [TLabel] isan individual marker (a string) pointing to a single granule of a certain granularity, e.g. "May-2002," "12-Jun-2004," "Christmas 2005," "9/11."

In Fig. 4, two sample temporal phrases together with their CG representation are presented, accompanied with examples of temporal relations, with which a CG may be augmented (for details see [16]).

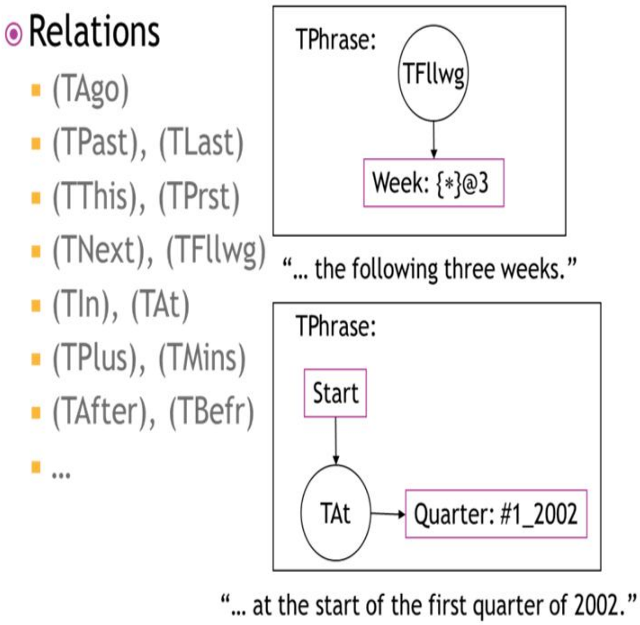

Figure 4. Sample temporal phrases and relations.Source: after [16]

In fig. 5, on the other hand, there are two sample temporal phrases augmented with temporal notions and indexes. Finally, in Fig. 6 a fully temporal conceptual graph, as presented by [16]is shown.

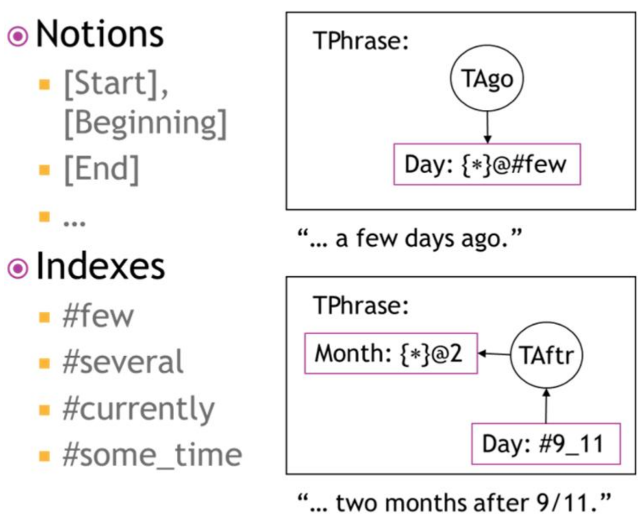

Figure 5. Sample temporal phrases, notions, and indexes.Source: after [16]

In next section we will show the use of temporal CGs for formalizing temporal knowledge about changes in enterprises' environment. We will use an example of barriers to entry to a market space.

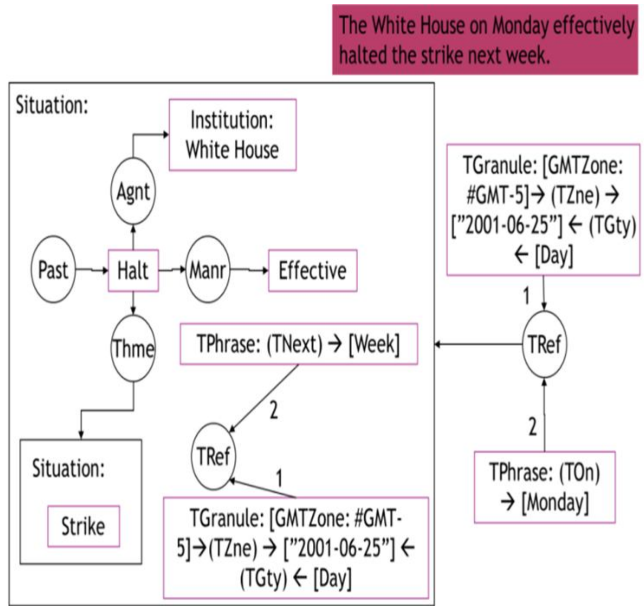

Figure 6. Example of a fully temporal conceptual graph.Source: after [16]

V. Economic Example: Barriers to Entry

In one of the best known models of enterprise's environment, by M. E. Porter[26], the environment consists of five forces: barriers to entry to a market space, the lobby of buyers, the lobby of suppliers, substitutes, and fight between market rivals. It is a very simplified model, nevertheless it shows that barriers to entry constitute one of the main factors that are to be analysed while formulating enterprise strategy.

Many definitions of barriers to entry can be found in the literature. The simplest and the most intuitive of them is the one saying, that a barrier to entry means everything that makes entry to a branch or market space difficult for economic entities [3].

What role do the entry barriers play in enterprise strategy? It is obvious that the knowledge on what kind of barriers our enterprise creates or could create, as well as the knowledge on barriers that can be possibly met by our firm, will significantly determine the process of elaborating corporate strategy.

The diversity of possible barriers to entry causes their importance as a problem or economic factor affecting enterprise operations. Moreover, barriers to entry affect operations (strategies) of both enterprises already operating in the market, which want to protect their market position, as well as enterprises planning to enter certain markets, while barriers may make the entry more difficult, cause a delay or even make entry impossible. Therefore, conducting an analysis of barriers to entry is a very important task. It becomes even more important if the analysis is performed dynamically, since barriers to entry – as each of the elements of enterprise environment – are not steady, but change in time. For example, the type of existing barriers changes, some of barriers appear and some disappear, also barriers' strength (height) changes.

TABLE I. SAMPLE VALUES OF VARIABLES DESCRIBING CAPITAL BARRIERS TO ENTRY.

TREF – TEMPORAL REFERENCE	DER – DOLLAR'S EXCHANGE RATE	OP – OIL PRICES
Past:	Sudden increase of DER	Sudden increase of OP
Over previous 3 months	Rapid increase of DER	Rapid increase of OP
Some time ago	Significant increase of DER	Significant increase of OP
Last week	Insignificant increase of DER	Insignificant increase of OP
On the recent Monday	Unexpected increase of DER	Unexpected increase of OP
Yesterday	Previously unexpected increase of DER	Previously unexpected increase of OP
A few days ago		
	Sudden decrease of DER	Sudden decrease of OP
Present:	Rapid decrease of DER	Rapid decrease of OP
In the current month	Significant decrease of DER	Significant decrease of OP
In the present quarter	Insignificant decrease of DER	Insignificant decrease of OP
This year	Unexpected decrease of DER	Unexpected decrease of OP
Today	Previously unexpected decrease of DER	Previously unexpected decrease of OP
Now		
	Low DER	Low OP
Future:	High DER	High OP
In the month to come	Stable DER	Stable OP
In the forthcoming elections	Unstable DER	Unstable OP
Will take place on Monday		
Next week	Sudden changes of DER	Sudden changes of OP
Soon	Rapid changes of DER	Rapid changes of OP
Over the next 3 days	Significant changes of DER	Significant changes of OP
Tomorrow	Insignificant changes of DER	Insignificant changes of OP
	Unexpected changes of DER	Unexpected changes of OP
Continuity:	Previously unexpected changes of DER	Previously unexpected changes of OP
Since/from the end of June	Frequent changes of DER	Frequent changes of OP
Over/during/for/within the next few days	Frequent fluctuations of DER	Frequent fluctuations of OP
In the past three weeks		
By/until/till the end of March 2003	Strengthening of dollar	
The following 3 weeks	Weakening of dollar	
Time stamp:		
As of May 2002		
In the first quarter of 2004		
On May 3, 2005		
On Tuesday		
In June		
Time fragments:		
At the beginning of/start of July		
From the end of 2004		
In mid/the middle of July		
Early/late in July		
Intervals/operators:		
From June 16 to July 10		
Between March and May		
In five weeks		
Two weeks after Christmas		
Three days later		
Five months before/back/earlier		

In what follows, we present temporal knowledge about two barriers to entry, namely dollar exchange rate and oil prices. Next we present sample temporal rules concerning these barriers, and their formalization with temporal CGs.

The rules have been formalized in the Prolog+CG language [15]. Prolog is one of the programming languages typically used in the AI, and the Prolog+CG version is a conceptual, object oriented extension of classical Prolog. It allows for using conceptual graphs as a basic data structure. For rule construction, three premise schemata were used:

1. Single premise: concerning either dollar exchange rate or oil prices,
2. A premise with conjunction of dollar exchange rate variable and oil price variable,
3. A premise with disjunction: dollar exchange rate variable or oil price variable.

Each premise contains also a temporal reference defining a temporal context of a rule.

Sample possible values of variables in rules' premises are shown in table 1.

Although oil prices and dollar exchange rate are both typical numeric variables, they are described in a qualitative way, such an approach does not exclude the possibility of a quantitative description.

It should be also pointed out that for this example a number of possible rules is 50.622[1], but in reality it may be infinite, because the number of temporal references in natural language

[1] The number of rules was calculated using the decision table theory, by multiplying the number of values of each attribute in a premise by other numbers of attributes' values.

DOI: 01.IJIT.3.3.5

is infinite. Therefore in practice it should be necessary to reduce the set of possible rules to a finite yet complete one.

Every rule is encoded in two versions:

1. a natural language version,
2. a Prolog+CG version, with conceptual graphs both in premise and in conclusion part.

Not to expand the length of the paper, below we present only a few sample rules. More rules are to be found in [20]. We use the following variables:

DER – dollar's exchange rate,

OP – oil prices,

Tref – temporal reference,

CRB – capital requirements barrier (to entry).

The sample rules are as follows.

1. IF over previous 3 months frequent fluctuations of DER occurred THEN CRB change proportionally

```
[CRB] – actn -> [change] – manr ->
[proportionally] :- [sit = [fluctuations]
– attr -> [frequent],
- ptnt -> [DER] ] <- tref – [TR = [[GCGty:
#ctx] – TAgo -> [month: {}@3]] <- TOvr].
```

2. IF in the month to come strengthening of dollar occurs OR OP remain high THEN CRB appear

```
a)[CRB] – actn -> [appear] :-
      [sit = [strengthening] – ptnt ->
[dollar]] <- tref – [TR = [month] - Tin -
> [GCGty: # to_come]].
b)[CRB] – actn -> [appear] :-
      [sit = [OP] – actn -> [remain]-
attr -> [high]] <- tref – [TR = [month] -
Tin -> [GCGty: # to_come]].
```

3. IF in the following 4 months quick changes of DER AND sudden decrease of OP occur THEN CRB lower

```
CRB] – actn -> [lower] :-
[sit1 = [changes] –
      - ptnt -> [DER],
      - attr -> [quick] ] <- Tref – [TR =
[[GCGty: #ctx] – Tin -> [Month: {}@4]] <-
Tovr ],
      [sit2 = [decrease] –
      - ptnt -> [OP],
      - attr -> [sudden]] <- tref - [TR =
[[GCGty: #ctx] – Tin -> [Month: {}@4]] <-
Tovr ].
```

4. IF in the first quarter of 2005 DER was stable AND OP were low THEN CRB were stable low

```
[CRB] – attr -> [stable],
      - attr -> [low],
      <- ptim - [past] :-
      [sit1 = [DER] – attr -> [stable] ]
<- tref – [TR = [Year: 2005] – TinOn ->
[Quarter: @1]],
      [sit2 = [OP] – attr -> [low] ] <-
tref – [TR = [Year: 2005] – TinOn ->
[Quarter: @1]].
```

The version based on conceptual graphs needed slight modifications in relation to the widely accepted CG notation, presented in Section 3, because of Prolog+CG language's particular requirements. They are as follows:

1. we could not use the AGNT relation, because, according to the formal requirements, only a living being may be an agent. Therefore, instead of writing e.g. [remain] – AGNT -> [OP], we used to write: [OP] – ACTN -> [REMAIN], where ACTN stands for action;
2. The premises are time-stamped (relation Tref, graph name: TR), while conclusions are not – there are only general references such as "past", "future", linked with the CRB variable by a PTIM relation. This is so, because the validity period of conclusions is not always the same as the validity period of premises. In our opinion, there is no justification for using the same time reference in premises and conclusions;
3. According to the Prolog+CG notation, relations' names are given without parenthesis;
4. According to the Prolog+CG notation, if the graph has several branches, they are separated by commas.

VI. Temporal Ontology For Economic Domain: Point Time and Calendar Time

As said in Section 4, one of the main elements of knowledge representation is ontology, because every logic – including temporal ones – assumes a domain ontology. Many authors defined ontology. Here we cite only a few definitions.

Eder and Koncilia[6] define ontology as a conceptualization of a domain. It describes domain notions, their properties and relations. A similar view of the ontology is presented by Salguero et al. [28], who describe it as a specification of knowledge domain conceptualization. A controlled dictionary depicting formally objects and relations between them, and has a grammar (p. 126).Grenon[8] writes that formal ontology is a branch of philosophy, that analyses and creates theories at the highest level of generality.

Perry et al. [24] add to the above that ontology assures the context or domain semantics (p. 147).

It seems that for time ontology the best definitions are these by [6] and [28]. It is because time ontology has to encompass time elements – points or intervals or both – and their relations. Moreover, it has to contain a „way" to manipulate these elements, so they become meaningful.

Knowledge representation of a dynamic economic environment should be composed of:

- Time and domain ontology,
- Computational procedures (reasoning)
- Temporal logic.

Time ontology refers to basic temporal entities and to time structure. If we assume, that time structure governs the choice of temporal entities, it becomes obvious, that the question of time structure is fundamental.

Time structure may be considered in many aspects, of which the most important concerns linearity, density and boundness/unboundness.

The first question concerning time structure is the question of time density (discreteness). Discrete time is perceived as a set of certain elements, while dense time – as a set of elements such that between any two elements there is always a third element. In other words, discrete time is modeled as a discrete set that may be identified as a set of integers, while dense time is modeled as a continuum – identified as a set of reals. Some authors, e.g. Galton, link the model of time as a set of integers or a set of reals only with dense time, and claim that dense time may be continuous (a set of reals) or non-continuous (a set of integers)[7].

The next problem concerns bounded *versus* unbounded time: it may be considered infinite in one or both directions from a certain point, labeled as "now".

Finally, the question of time linearity or nonlinearity. The most widely accepted model of time is linear. There are also models of non-linear time, such as: time branched in the future, time branched in the past, time branched in both directions (parallel time) or circular time. The motivation for solutions with branched time was an assumption that many different pasts ("ways") might have led to "now", and – respectively – many different futures ("ways to the future") may start "now".

Up till now, research on temporal reasoning focused mainly and the most often on linear models of time (e.g. Allen's works). For more complex tasks, however, such as analysis of distributed systems, or cooperative robots programming, also more complex models were considered, e.g. partially ordered time. More details on time structures and their applications can be found in [11].

What time structure would be suitable for economic tasks? For example for the task of dynamic analysis of barriers to entry, presented in Section 5. First let us recall the most important features of economic environment in context of temporal representation. Those features are as follows:

a) Changes of the economic environment may be both continuous and discrete. Also environment's characteristics can be treated as continuous or discrete features. Kotler[18]lists the elements of the environment (pp. 698-700). It can be immediately seen that they are either continuous or discrete;
a) Changes of the economic environment can be easily placed on time axis, and in the simplest case this would be calendar time axis;
a) The time horizon of those changes can be viewed as infinite in both directions starting from the point labeled "now". Nevertheless, from the practical point of view, it is justified to limit the time horizon of the analysis in the past.

Taking into account the above mentioned features of the economic environment, we claim that for the analytical task, formulated above, we may assume time to be discrete, left linear, infinite in the future, but finite in the past. We decided to choose such a structure of time for the following reasons:

a) point (discrete) time – in our opinion discrete time structure is adequate for the task, although in practice many of the elements in the environment change continuously. Some authors even postulate a process of continuous monitoring of environment. However, some barriers to entry (e.g. legal ones) change in a discrete way. Moreover, from the practical point of view, it is not possible to "feed" the intelligent analytical system with the information continuously. Thereforechanges have to be recorded in a discrete way. It would therefore seem more natural and more justified to choose a discrete time structure.
b) Left linear time (time branching into the future) – it seems that for the temporal analysis of environment such a structure is quite sufficient. Assuming a structure of time branching into the future, that is a structure in which present activities may result with many different future scenarios, would deepen the analysis, allowing e.g. to perform forecasts "what…if". And taking into account the differences between temporal aspects of different markets, it would be worthy of attention whether parallel time structure would be suitable for analytical tasks. In our opinion – yes, as it would enable to perform the analysis on many markets at the same time. A structure of time branching in the past may also be considered – as a framework for analysing, which past changes on different parallel markets are responsible for the present situation of an enterprise. The question of using non-linear time structures in economic analysis performed by an intelligent system is undoubtedly a very interesting research problem;
c) Time infinite in the future – such structure seems obvious: at a given moment it is not possible to define, for how long an enterprise would exist, and – in consequence – for how long the analytical process will be performed. Of course we may limit the time horizon for the analysis, assuming for example, that we are interested only in 5-years perspective. But in general, the time horizon depends on the feature being analyzed, therefore it is not purposeful to limit the perspective and we claim that assuming time unlimited in the future is justified. Also justified in our opinion is assuming time limited in the past: nor an enterprise, nor an intelligent system exist "from always", and the analysis has not been performed for hundreds of years. Moreover, such analysis would not be possibly useful, especially having in mind turbulences and discontinuity of economic environment. We are convinced that it is advisable to assume a certain "past time horizon" for the analysis, so limiting time in the past is purposeful and justified.

Formally, a time structure T is linear, if:

$$\forall t_1, t_2 \in T: (t_1 < t_2) \vee (t_1 = t_2) \vee (t_2 < t_1).$$

Time structure T is left-linear (i.e. branching into the future) if

$$\forall t_1, t_2, t_3 \in T\ (t_2 < t_1 \wedge t_3 < t_1) \Rightarrow (t_2 < t_3) \vee (t_2 = t_3) \vee (t_3 < t_2).$$

Time structure T is right-linear (i.e. branching into the past) if

$$\forall t_1, t_2, t_3 \in T\ (t_1 < t_2 \wedge t_1 < t_3) \Rightarrow (t_2 < t_3) \vee (t_2 = t_3) \vee (t_3 < t_2).$$

Finally, a time structure T is a parallel one, if it is at the same time left- and right-linear, that is branching in both directions.

Fore conomic analysis, we assume a point time. That is, time consistsof points and of a precedence relation <, i.e. a

point structure T is an ordered pair $\langle T, < \rangle$, where T – a nonempty set of points, and < - precedence relation.

The axiom of time discretness may be written as:

- $\forall x, y\, (x < y \rightarrow \exists z\, (x < z \,\&\, \neg\exists u\, (x < u \,\&\, u < z\,)))$
- $\forall x, y\, (x < y \rightarrow \exists z\, (z < y \,\&\, \neg\exists u\, (z < u \,\&\, u < y\,)))$

This structure has the following properties:

- transivity $\forall x, y\, (x < y \,\&\, y < z \rightarrow x < z)$
- anti-reflexivity $\forall x\, \neg(x < x)$
- anti-symmetry $\forall x, y\, (x < y \rightarrow \neg\, (x < y))$

Moreover, as we deal with time branching into the future (left-linear), we add an axiom of backward linearity:

$$\forall x, y\, (x < z \,\&\, y < z \rightarrow X < y \vee x = y \vee y < x).$$

The point time ontology presented in this section extends and modifies the ontology by Hobbs and Pan [13], also adding time properties from [11], because of economic application domain, presented before.

Time points are a subclass of temporal entities:

$$Instant(t) \rightarrow TemporalEntity(t)$$

$$\forall (T)\; TemporalEntity(T) \rightarrow Instant(T)$$

Predicates *begins* and *ends* are the relations between points and temporal entities:

$$begins(t,T) \rightarrow Instant(t) \wedge TemporalEntity(T)$$

$$ends(t,T) \rightarrow Instant(t) \wedge TemporalEntity(T)$$

Moreover

$$Instant(t) \equiv begins(t,t)$$

$$Instant(t) \equiv ends(t,t).$$

If exists a beginning and an end of a temporal being, it is unique:

$$TemporalEntity(T) \wedge begins(t_1, T) \wedge begins(t_2, T) \rightarrow t_1 = t_2$$

$$TemporalEntity(T) \wedge ends(t_1, T) \wedge ends(t_2, T) \rightarrow t_1 = t_2$$

Predicate *TimeBetween*is a relation between a temporal being and two points:

$$TimeBetween(T, t_1, t_2) \rightarrow TemporalEntity(T) \wedge Instant(t_1) \wedge Instant(t_2) \wedge (Instant(t_1) < Instant(t_2) \vee Instant(t_2) < Instant(t_1) \vee Instant(t_1) = Instant(t_2))$$

The condition in the above expression - $(Instant(t_1) < Instant(t_2) \vee Instant(t_2) < Instant(t_1) \vee Instant(t_1) = Instant(t_2))$ – means that time points lie on the same time branch. It is necessary to add this condition, because only on the same time branch (time axis) the condition of strict linear order is fulfilled, which allows to compute the value of *TimeBetween* predicate. It is not possible to compare distances between points on different time branches.

It should be discussed how time is linked to events in the world. Hobbs and Pan propose to use 4 predicates: *atTime, during, holds, timeSpan*[13] p. 70. We extend here the notion of an event. The classic definition says that an event is a dynamic picture of the world, causing changes in facts. Following Hobbs and Pan, we will however understand events very broadly – as "anything that may be placed in time" ([13] p. 70), so not only event by itself, but also a state, a process, a logical statement etc.

As said above, Hobbs and Pan propose four predicates. Because we adopt a discrete model of time, we do not use predicate *during*, concerning intervals, and the predicate *timeSpan* will have a narrower definition compared to the original one (see *ibid.*, p. 71).

Predicate *atTime* linksan event with a time point, therefore it is crucial for the ontology of discrete time. It says that an event happens, occurs at time point *t*.

$$atTime(e,t) \rightarrow Instant(t)$$

Predicate *Holds* is generally a duplicate of predicate *atTime*. In the original approach by Hobbs and Pan it says that an event occurs at time point *t* or over a time interval *T*. As we assume time discreteness, we omit the second meaning of the predicate and in this way we duplicate the two predicates. We may write: $holds(e,t) \equiv atTime(e,t)$.

Finally, the predicate *timeSpan* links events with time points (or sequences of time points) – it is a narrowed version of the original predicate by Hobbs and Pan, which linked events also with intervals and sequences of intervals. This predicate is used for states or processes that adhere to each other, it shows the whole time span during which a process or a state holds. Formally:

$$timeSpan(T,e) \rightarrow TemporalEntity(T) \vee tseq(T)$$

where $tseq(T)$ is a sequence of time points. Moreover

$$timeSpan(t,e) \wedge Instant(t) \rightarrow atTime(e,t)$$

$$timeSpan(t,e) \wedge Instant(t) \wedge t_1 \neq t \rightarrow \neg\, atTime(e, t_1)$$

The predicate *atTime* links an event with a concrete time point, but this is not a direct linking of an event with the date of its occurrence. At the same time, dates are necessary in the description of economic reality. Therefore there is a question how to link time branching into the future with calendar time.

As McDermott pointed out [21], two dates cannot be placed on two different time branches, but one date (the same one) may be placed on many branches, as time branches are independent. Therefore in McDermott's opinion one should discuss a date line independent from the main time structure. In this way, the date line preserves a linear order.

If we adopt the solution proposed by McDermott, we will have to extend the time structure presented earlier to the following one[3]:

$$T = T, D, <_{tt}, <_{dd}, <_{td}, <_{dt} \rangle$$

where T – a set of time points, D – a set of dates, $<_{tt}$ – backward partial linear order over T, $<_{dd}$ – a linear order over D, $<_{td}$ and $<_{dt}$ are precedence relations linking the former two orders. In this situation we need to add a few new axioms to the ones presented in section 3. Hajnicz calls them the axioms of quasi-transitivity:

$$t_1 <_{tt} t_2 \wedge t_2 <_{td} d \rightarrow t_1 <_{td} d$$

$$d_1 <_{dd} d_2 \wedge d_2 <_{dt} t \rightarrow d_1 <_{dt} t$$

$$d <_{dt} t_1 \wedge t_1 <_{tt} t_2 \rightarrow d <_{dt} t_2$$

$$d_1 <_{dt} t\;\, t <_{td} d_2 \rightarrow d_1 <_{dd} d_2$$

$$t <_{td} d_1 \wedge d_1 <_{dd} d_2 \rightarrow t <_{td} d_2$$

$$t_1 <_{td} d\;\, d <_{dt} t_2 \rightarrow t_1 \neq t_2 \wedge \neg\, (t_2 <_{tt} t_1)$$

Adopting the extended structure of time and additional axioms, we have a time theory that is described by the notions of transitivity, anti-symmetry, backward linearity and quasi-transitivity. Together with the ontology of point left linear time, we are able to place economic events in time.

[3] The solution presented in this section comes from [11], p. 24-25.

DOI: 01.IJIT.3.3.5

VII. Ontology Representation in RDF and OWL

Ontology is one of the main components of the Semantic Web. The latter is defined as adding semantic mechanisms to the existing www network, in order to define information and to enable a better cooperation between humans and computers. Semantic Web may be also seen as composed of several main elements:

- Syntactic basis: XML
- Basic data model for facts, and rules of creating simple ontologies: RDF and RDF Schema
- Ontological vocabulary: more expressive languages and W3C standards (OWL)
- Logic: ontologies, vocabularies, procedural rules,
- Proofs: reasoning, information exchange, validation
- Confidence: digital signatures, recommendations, safety

RDF stands for Resource Description Framework. Its definition and standard may be found e.g. at http://www.w3.org/RDF/ and http://www.w3.org/TR/rdf-primer/ The main assumption underlying RDF is that everything is a resource. The resource in turn is everything that may be identified by an URI – Uniform Resource Identifier. The statements about resources are in the form of so-called RDF triplets, composed of subject, verb and complement. Other forms of RDF triplets are: subject, predicate, object, and object, property, value. The set of triplets together forms a RDF model. Therefore a RDF document is a description of resources and relations between them. The RDF Schema defines resource classes, types of relationships and of constraints, thus enabling the process of reasoning.

The RDF language has many applications, among others:

- Description of features of purchased goods,
- Description of a schedule of events in the web,
- Presentation of information on web pages,
- Creation of a document or its modification,
- Description of images,
- Description of contents of search systems,
- Description of digital libraries.

OWL (Web Ontology Language) has much broader possibilities of depicting relationships among classes and of reasoning. The standard may be found at http://www.w3.org/2004/OWL/ and it is recommended by the W3C. OWL has been developed to describe and exchange knowledge over the web. It is based on Description Logic and defined by the W3C as three different sublanguages:

- OWL Full
- OWL DL
- OWL Lite

Each of these sublanguages covers different requirements. OWL Full uses all constructs of OWL syntax. It is fully compatible up with RDF, both syntactically and semantically. It may be combined with RDF and RDF Schema constructs. A kind of drawback of OWL Full is that there is no full support for reasoning. Next, OWL DL conforms to Description Logic. Not all constructs from OWL and RDF may be used with this version. It is neither fully compatible with RDF – not every proper RDF document is a proper OWL DL document, but every proper OWL DL document is a proper RDF one. Also, OWL DL supports effective reasoning. Finally, OWL Lite is a subset of OWL DL constructs. It is the easiest of the three sublanguages in context of understandability and implementation. A drawback of this sublanguage is its limited expressiveness.

Both RDF and OWL languages are not suited for representing temporal knowledge. Therefore in the literature appeared some attempts to solve this problem. For example, the authors of [9] proposed temporal RDF graphs. In [24] a spatio-temporal data model, adapting temporal RDF graphs is presented. As for OWL, a new OWL-based temporal formalism (TOWL) is presented in [22], and a spatio-temporal OWL (STOWL) is introduced in [28].

VIII. Reasoning: Temporal Reasoning and the TAL Language

Generally speaking, there are two approaches to temporal reasoning: a model-based one and a temporal languages one. The temporal languages are those dedicated to description of changing realm, and using techniques of automated reasoning [4]. The first approach is represented by e.g. time series and their use to represent time-variant variables (see e.g.[12]). It is not the topic of our considerations. The second approach will be presented below on the example of the TAL language (*Temporal Action Language*). We will present a practical example, concerning a problem of establishing, whether an unemployed person may be granted a benefit. The example chosen is very simple, to focus attention on a temporal languages approach.

The TAL (Temporal Action Language) is derived from Sandewall's PMON logic [5]. Its main features as a language for describing temporal dependencies include: the notion of time independent from actions, the possibility of defining causal dependencies apart from actions' definitions, and the possibility of describing concurrent interactions.

The language consists of two levels (layers): the so-called surface language, which is used to describe narratives (for more information see [5]), and the so-called base language, namely the logic of events, which is an ordered 1st order predicate logic. Any correct narrative description, after being transformed into the description in the base language constitutes a finite set of 1st order wffs.

The surface language layer consists of:

- temporal expressions,
- value expressions,
- atomic expressions,
- narrative statements,
- additional macro-operators and abbreviations.

The base language layer (the logic of events) contains, among others, temporal predicates: HOLDS, OCCURS, OBSERVE, DUR, PER and others (the exact definitions of the predicates can be found e.g. in [5].

The whole formal inference process is conducted after "translating" the description in the surface language into the description in the base language.

ACEEE Int. J. on Information Technology , Vol. 3, No. 3, Sept 2013

The example will be illustrated with Canadian unemployment law. The illustration comes from [25]. The authors present there a concrete decision problem, which in our opinion could be solved by using the TAL language. All the rules come directly, or after slight modifications, from Canadian unemployment law.

The example to be discussed is presented in Fig. 7.

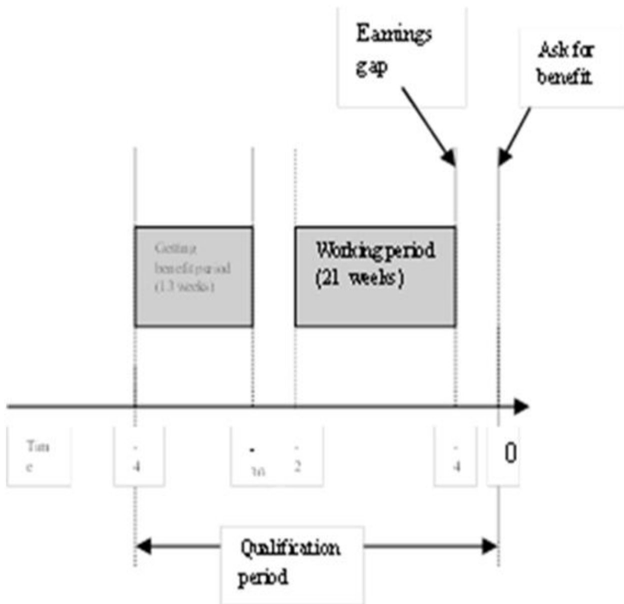

Figure 7. Sample decision situation.
Source: own elaboration based on [25].

As we can see in Fig. 7, a person asking for benefit lost work (in a week numbered with -4) and asked for benefit in a week numbered with 0. The former working period lasted for 21 weeks. The person had already been granted a benefit, and has been getting it for 13 weeks. For our purposes it is not important, what happened between weeks -25 and -30. We may assume that it was a working period or an unemployment period. The most important information concerning working period is that it lasted for minimum 21 weeks: on this basis we may establish, whether the person qualifies for a new benefit (this results from the Canadian law).The basic problem (as in original work[25]) is to establish, whether the person has the right to the new benefit period. In our example the answer is yes, because the former working period was longer than 20 weeks (more details about the legal rules are to be found in [25]). It is apparently a temporal information. Nextwe have to establish, how long a new benefit period is tobe. This in turn depends on the information about theformer benefit period, because the starting point of so-called qualification period is the point in which the former benefit period started. As it can be seen in the figure, the qualification period for the sample person is 43 weeks.

Summing up – the temporal aspect of the problem concerns establishing the longitude of qualification period and – in consequence – the new benefit period of an unemployed person.

The tool for the implementation of the problem is the TAL language and its implementation named VITAL. The main task for VITAL was to calculate the longitude of qualification period, needed for establishing, whether a person will be granted a benefit, or not. As input data the following information has been provided:

- The fact, that the person has already been granted a benefit in the past (and for how many weeks),
- The formerworking period,
- A point time in which a person asked for the new benefit.

It should be pointed out here, that the assumed time granularity is one week, therefore time point is the number of a week.

We have encoded input data in a scenario written in the TAL language, in form of so-called occurrences. From these occurrences the VITAL tool was to infer the longitude of a qualification period. To enable this inference, we had also to encode a proper temporal dependency, linking input facts with the way of calculating the qualification period. The dependency has been taken from the Canadian unemployment law.

A key to success (that is to establish the longitude of qualification period) is a rule stating, that qualification period starts in the same time point as the former benefit period, and ends in the time point in which a person asks for a new benefit. Therefore, in terms of the TAL scenario, if we denote by *t1* the time point in which the variable *getting_benefit_1* becomes true, and by *t2* – the time point in which action *ask* (asking for new benefit) becomes active, then the variable *qual_period* (denoting the qualification period) should become true (the system should infer this value) over the interval [t1, t2). The discussed dependency, has the following form in the TAL language:

```
depforall t1, t2 [
      Ct([t1] getting_benefit_1) &
      Ct([t2] ask) &
      !exists t3 [t1 <= t3 & t3 < t2
&Ct([t3] ask) ] ->
            I([t1, t2) qual_period)]
```

where:
dep – dependency label,
t1, t2 – time points,
Ct – operator *changes to true*,
I – macrooperator, that assigns a new value to the default value of a feature, over a particular time interval([5]).

The dependency stated as above will work also in a situation, when the same person will be hired again, then will get benefit again (*getting_benefit_1*), and then will again ask for benefit (*ask*). In such a situation the variable *qual_period* will be true from the first moment in which *getting_benefit_1* becomes true, up till the last action of asking for benefit. By formulating the dependency in the form presented above, we may check, whether *t2* is the shortest time point after *t1* such, that Ct([t2] ask).

The sample situation presented in Fig. 7 has been encoded in the VITAL tool. The only difference was that while numbering time points (weeks), positive numbers have been used. The result of temporal reasoning performed by the

DOI: 01.IJIT.3.3.5

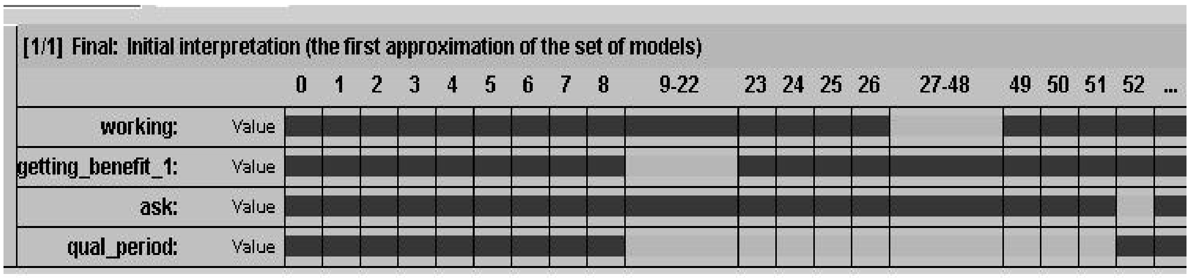

Fig. 8. The result of calculating the qualification period in the VITAL application. The lighter color on the timeline denotes the true value of a feature, while the darker one denotes the false value (in reality the drawings prepared by the VITAL tool are colorful). Source: own elaboration. Working - the time person has been working, other notations as in the text of the paragraph.

VITAL tool is presented in Fig. 8.

What is known after performing temporal reasoning by the VITAL tool? It is known for how many weeks the person has been getting the former benefit (if at all; in our example – for 13 weeks), for how long the person has been employed during the qualification period (from this information it will be possible to infer if the person may be granted the next benefit; in our example the working period is 21 weeks), how long is the qualification period (in our example – 43 weeks). All this information can be read from the above figure.

Moreover, we may assume that while asking for benefit, a person is obliged to provide some additional information, e.g. on family situation, education, additional skills etc., to enable decision on other forms of help. It makes no sense encoding this information in the VITAL tool, as it is not of temporal kind. Nevertheless the results of reasoning performed on temporal data, together with the additional information, may constitute an input for further, non-temporal reasoning.

IX. Temporal Reasoners - Reasoners For Temporal Ontologies/ Knowledge Bases: Proton

In Sections 2 and 3 we defined a temporal intelligent system. To the best of our knowledge, there are no functioning systems of that kind. Instead, there are implemented systems for managing temporal information, using single temporal or semi-temporal formalisms. One of them is a tool called PROTON, a reasoner acting over temporal ontologies. We will briefly discuss it here, on the basis of [23].

PROTON stands for “Prolog Reasoner for Temporal Ontologies in OWL”. It is a reasoning system for managing temporal information over OWL ontologies. The construction of the tool adopts a 4d-fluent approach to representing temporal information in ontologies. It is possible to represent time points or time intervals, and also events occurring in these time points or intervals.

Behind the construction of PROTON, there are some observations and assumptions. First observation is that such representation languages, as OWL and RDF, are based on binary relations. The temporal information, that is an information with explicit temporal aspects, may be represented in ontologies, but its changes cannot be represented in OWL nor RDF. As a consequence, while performing reasoning over information represented in OWL or in RDF, one cannot take the temporal aspect of information into account.

PROTON is dedicated to reasoning about temporal ontologies in OWL, and is capable of handling queries concerning events changing in time. It starts with a temporal ontology in OWL, then transforms it into triplets of the form (subject predicate subject). Finally, these triplets are transformed into Prolog clauses. PROTON is implemented in so-called temporal situation calculus, thus it can make use of Prolog mechanisms for implementing the reasoning component.

The main formalisms for reasoning about actions and change are: situation calculus, fluent calculus, event calculus, action languages, action calculus,and temporal action logic (TAL) which has been presented above. Situation calculus is the most popular of them, a second order language aimed at representing changes of the world. In SC, all changes result from performing some actions. The world itself is depicted using so-called fluents (which may be predicates or functions). A probable evolution of the world (domain) is a sequence of actions represented with situations.

The only reason for using temporal situation calculus in PROTON is – in our opinion – the need for using Prolog reasoning, with which SC is strictly connected. Nevertheless, the authors of PROTON propose their own extension to classic SC in order to incorporate time notion explicitly. Their modifications are as follows[23]:

- For each fluent f, an argument L is added, where L is a list of time intervals $[a; b]; a < b$.
- Each $[a; b]$ represents time points x: $\{x\{a < x < b\}\}$.
- Fluent f is true over all intervals from list L.
- Two new functions have been defined: $start(a)$ and $end(a)$, where a is an event (action).
- Events are ordered: $a1 < a2$, where $start(a1) < start(a2)$.
- Predicate $eventHappen(a; t)$ means, that action a is executed at time moment t.
- A so-called “temporal situation” is defined as a situation with a list of time intervals, over which fluents are true.
- Function $Holding(S; t)$ returns all fluents true at time t. For a functional fluent, $Holding(S; t)$ returns a value of the function in point t. Situation S is a temporal one.
- Passing from situation to situation is possible when $Holding(S; t)$ returns different sets.

What is immediately seen after this short list, is a strange way of representing time points. In our opinion a point should be represented as an interval {a,b} where a=b. This is the

ACEEE Int. J. on Information Technology , Vol. 3, No. 3, Sept 2013

way Allen represents time points (see[1]). This is the more justified, that the authors of PROTON later on adopt Allen's interval calculus for manipulating time intervals.

Next, in our opinion the authors needlessly multiply temporal entities, introducing both time points and time moments. Time points would be quite enough. Of course, one may suppose that for the authors time points and time moments mean the same, but this is not stated anywhere in the paper cited. PROTON tool is composed of several modules: SWI-Prolog, which converts OWL concepts into Prolog clauses, a module for calculating interval relations, a set of functions for calculating property values in points of time, a set of predicates establishing, when an event takes place, and finally, a set of rule executing predicates, which operate when an event or a change in feature value occurs.

OWL concepts are transformed into facts of the form "predicate(subject, object)". Before this can be done, the OWL ontology is converted into triplets (subject predicate object) with SWI-Prolog. After transforming all OWL objects into facts it is possible to use Prolog reasoning for implementing the reasoner. PROTON's knowledge base consists of automatically created predicates, concerning the domain of application. Some of them are common for all ontologies, while some are domain-specific. Also implemented are Allen's temporal relations (before, equals, meets, overlaps, during, starts, finishes). There are some additional predicates in the knowledge base –two predicates for time handling, two predicates for calculating feature values – domain dependent, and some other predicates, among which there are those responsible for solving the frame problem.

There are some questions arising while analyzing PROTON's structure and knowledge base. First, it seems strange to put predicates into the KB, while at the same time these predicates "scan" the KB and establish some values. It seem that these are functions, not predicates, the question is why the authors of [23] call them predicates?

Second, in our opinion there is no need to introduce both time points and time intervals, if the relations in the KB are those introduced by Allen (as commonly known, he introduced only interval relations and has shown how to define time points in terms of intervals). It would be quite enough then to introduce only intervals of time into PROTON's knowledge base.

The authors do not explain, how past events are handled and how is the KB actualized? And last but not least, they write about a knowledge base in one place, while about database in other. In this way the reader does not finally know, whether PROTON contains a KB or a DB or both.

Summing up, PROTON is one of the first attempts to implementation of a temporal reasoner over OWL ontologies, not free from weaknesses.

Some other interesting temporal reasoners are e.g. SOWL[2],FuzzyTIME temporal reasoner[14] and others.

X. Conclusions

The domain of temporal reasoning used for economic and managerial problems is vast and it is not possible to cover all the questions in one paper. We tried to show a broad survey of problems, focusing on the triple formulated by Sowa [29]: logic, ontology and computational models, which together form a knowledge representation of a domain. As the economic domain is dynamic and time is a crucial aspect of it, we presented questions linked with temporal logic, temporal ontologies, and temporal computations – that is, questions of temporal reasoning for dynamic managerial decisions.

Of course, the solutions presented in the paper are not the only possible ones. We only found them representative for the problem, and interesting.

The temporal economic knowledge is a specific one, therefore it needs a specific treatment. The solutions presented in the paper do not account for a so-called commonsense reasoning, which may be very useful for formalizing economic decisions and/or economic expertise. Therefore we also plan to present a survey of commonsense reasoning solutions which may be applied to economic and managerial problems.

References

[1] Allen J. F., Towards a General Theory of Action and Time. „Artificial Intelligence", Vol. 23 No. 2, July 1984.

[2] Batsakis S., Petrakis E. G.M., SOWL: A Framework for Handling Spatio-Temporal Information in OWL 2.0, in: Rule-Based Reasoning, Programming, and Applications: 5th International Symposium, RuleML 2011 - Europe, Barcelona, Spain, July 19-21, 2011, Eds. Nick Bassiliades, Guido Governatori and Adrian Paschke (Sep 28, 2011), Springer, pp. 242-249.

[3] Barriers to Entry and Exit in UK Competition Policy. A Report by London Economics for the Office of Fair Trading. OFTRP2, March 1994.

[4] Bouzid M., Ligęza A., Temporal Logic Based on Characteristic Functions. Proc. KI-95: Advances in Artificial Intelligence, 19th Annual German Conference on Artificial Intelligence, September 1995. Springer.

[5] Doherty P., Gustafsson J., Karlsson L., Kvarnström J., Temporal Action Logics (TAL): Language Specification and Tutorial. Linköping Electronic Articles in Computer and Information Science Vol. 3 (1998): nr 015. http://www.ep.liu.se/ea/cis/1998/015. October 1, 1998.

[6] Eder J., Koncilia Ch., Interoperability in Temporal Ontologies. Proc. Of the Open Interop Workshop on Enterprise Modeling and Ontologies for Interoperability (EMOI-INTEROP'05), June 13-14, 2005, Porto, Portugal; ceur-WS.org, access: 04.11.2010

[7] Galton A., Time and Change for AI, In: Gabbay D. M., Hogger C. J., Robinson J. A. (Eds.), Handbook of Logic in Artificial Intelligence and Logic Programming, Volume 4: Epistemic and Temporal Reasoning. Clarendon Press, Oxford 1995.

[8] Grenon P., The Formal Ontology of Spatio-Temporal Reality and its Formalization. AAAI Technical Report SS-03-03. www.aaai.org, access: 04.11.2010

[9] Gutierrez C., Hurtado C., Vaisman A., Temporal RDF. Proc. European Conf. on the Semantic Web, Crete, Greece, 2005.

[10] Hajnicz E. Logical representation of time-varying knowledge. PLJ, Warszawa 1996 (in Polish).

[11] Hajnicz E., Time Structures. Formal Description and Algorithmic Representation. LNAI 1047, Springer, 1996.

DOI: 01.IJIT.3.3. 5

[12]Hauke K., Formalization of dynamic knowledge using selected statistical methods. In: Baborski A. (ed.), Knowledge acquisition in databases.Research papersAE Wrocław nr 891, Wrocław 2001 (in Polish).

[13] Hobbs J. R., Pan F., An Ontology of Time for the Semantic Web. "ACM Transactions on Asian Language Information Processing", Vol. 3, No. 1, March 2004, pp. 66-85.

[14] Juarez J. M., Campos M., Morales A., Palma J., Marin R., Applications of Temporal Reasoning to Intensive Care Units, "Journal of Healthcare Engineering" Volume 1, Number 4 / December 2010, pp. 615-636.

[15] Kabbaj A., Prolog+CG User's Manual. Version 2.0., 01-12-2000.www.insea.ac.ma/CGTools/PROLOG+CG.htm access: 30 July 2005.

[16] Kalczynski P., Temporal Semantic Networks for Business News. University of Toledo Research Paper, 2005.

[17] Kaleta A., Competition strategy for industry. Wydawnictwo Akademii Ekonomicznej im. O. Langego we Wrocławiu, Wrocław 2000 (in Polish).

[18] Kotler P., Marketing. Analysis, planning, implementation, and control. Gebethner&Ska, Warszawa 1994 (in Polish).

[19] Kvarnström J., Doherty P., VITAL research tool, 1997. Available at: http://anton.ida.liu.se/vital/vital.html

[20] Mach M. A.,Temporal analysis of enterprise's environment. Intelligent tools and techniques. AE, Wrocław, 2007 (in Polish).

[21] McDermott D., A Temporal Logic for Reasoning about Processes and Plans. "Cognitive Science" Vol. 6, 1982, pp. 101-155.

[22] Milea V., Frasincar F., Kaymak U., Knowledge Engineering in a Temporal Semantic Web Context, Eighth International Conference on Web Engineering, IEEE 2008, pp. 65-74.

[23] Papadakis N. ,Stravoskoufos K., Baratis E., Petrakis E., Plexousakis D., PROTON: A Prolog Reasoner for Temporal Ontologies in OWL. "Expert Systems with Applications" Volume 38, Issue 12, November–December 2011, Pages 14660–14667. http://www.sciencedirect.com/science/article/pii/S0957417411008001 access: 26.10.2012

[24] Perry M., Hakimpour F., Sheth A., Analyzing Theme, Space, and Time: An Ontology-based Approach. Proc. 14th annual ACM international Symposium on Advances in geographic information systems (GIS-06). ACM, New York, NY, USA, 2006, pp. 147-154.

[25] Poulin D., Mackaay E., Bratley P., Frémont J., Time Server – a legal time specialist. Proc. Third International Conference on Logic, Informatics, Law. Florence, 2-5 November 1989.

[26] Porter M. E., Competition strategy. Methods for analysing sectors and competitors. PWE, Warszawa 1999 (in Polish).

[27] Reiter R., Knowledge in Action: Logical Foundations for Describing and Implementing Dynamical Systems, Version of May 1, 2000. http://www.cs.toronto.edu/cogrobo

[28] Salguero A., Araque F., Delgado C., Spatio-temporal ontology based model for Data Warehousing. 7th WSEAS International Conference on Telecommunications and Informatics (TELE-INFO'08), Istanbul, Turkey, May 27-30, 2008, pp. 125-130

[29]J. Sowa, „Knowledge Representation" Pacific Grove, CA: Brooks/Cole 2000.